Calibration of Head-Mounted Displays for Augmented Reality Applications

ADAM L. JANIN, DAVID W. MIZELL AND THOMAS P. CAUDELL
Research and Technology
Boeing Computer Services
MS 7L-48
P.O. Box 24346
Seattle, WA 98124-0346
janin@espresso.boeing.com (email)

Abstract

We have been developing "Augmented Reality" technology, consisting of a see-through head-mounted display, a robust, accurate position/orientation sensor, and their supporting electronics and software. Our primary goal is to apply this technology to touch labor manufacturing processes, enabling a factory worker to view index markings or instructions as if they were painted on the surface of a workpiece. In order to accurately project graphics onto specific points of a workpiece, it is necessary to have the coordinates of the workpiece, the display's virtual screen, the position sensor, and the user's eyes in the same coordinate system. We describe the linear transformation and projection of each point to be displayed from world coordinates to virtual screen coordinates, and then characterize the experimental procedures for determining the correct values of the calibration parameters.

Introduction

Most Virtual Reality research now in progress centers around the use of "immersive" head-mounted displays (HMDs) -- those which occlude the user's view of the actual physical surroundings, so that 100% of what the user sees is computer graphics. The earliest experiments in VR, however [Sutherland 68], made use of a transparent HMD and a position sensing system, enabling the displayed graphics to appear to be located in a fixed position in space. These prototyping efforts in see-through HMDs evolved into work on military pilot situation displays, which by now see wide use [Furness]. At Boeing, we are developing see-through HMD technology and associated components such as position sensors with the goal of applying see-through display systems to touch labor manufacturing and assembly processes. We have termed this technology "Augmented Reality" [Caudell, Mizell], to signify that the HMD is transparent and the user sees his actual surroundings with the addition of some computer-generated graphics.

Components of a working Augmented Reality (AR) system include a see-through HMD, a position and orientation sensing system, and support electronics and software. The basic concept underlying the application of AR to touch labor manufacturing is the following: if the display, the user's eyes, the position/orientation sensor, and the workpiece can all be located in the same coordinate system, then the display can "project" simple, line-oriented graphics or text onto the surface of the workpiece (the focal distance of the display is set at approximately arm's length, the distance at which the worker is assumed to be viewing the workpiece). Because the head's position and orientation is frequently sensed, the superimposed information can be stabilized on specific coordinates on the workpiece's surface. The result is that the graphics appear to the user as if they were painted on the surface of the workpiece.

Aircraft manufacture and assembly include many manual processes. Furthermore, modern commercial transport aircraft are complex designs, and each Boeing aircraft coming down the line differs significantly from the one before it. Each may be configured for a different customer or a different set of routes. There are many steps in the process where workers have to check a

0-7803-1363-1/93 $3.00 © 1993 IEEE

drawing, superimpose a template, or otherwise obtain graphical information about the manufacturing or assembly step they are about to perform. If AR technology could be applied, this information would appear as if it were painted on the workpiece at hand. As the worker finished each step, a voice command or button push would cause the system to bring up the graphics associated with the subsequent step of the process.

Key R&D Issues

Below we list the key research and development issues that must be addressed before the practical application of AR to touch labor manufacturing can be realized. In general, it should be pointed out that AR raises the challenge of orthostereoscopy [Robinett & Rolland] to a new level. Typically, the requirement of a VR display and position sensor is only that the generated scene look realistic to the user. For AR, the demand is that the displayed graphics match up, for some applications quite accurately, against specific points in the user's real surroundings. It is also worth noting that in some cases, the most difficult issue is not how to solve the technical problem, but rather how to deploy systems cheaply enough so that AR becomes a practical alternative.

1. Fortunately, AR requires much less computational capability than VR. Instead of colored, shaded solid geometry as in VR, we typically only render 30 or 40 monochrome lines per frame. Thus, the design of the on-board computing electronics poses no fundamental research problems but should be carefully engineered for cost-effectiveness and performance. We desire a belt- or back-mounted, untethered onboard system with sufficient performance to compute the geometric transforms and drive a stereo display at 20 or more frames per second. It may also need to perform some processing of the position/orientation sensor's raw inputs.

2. The design of the see-through HMD is critical. We want a light, comfortable, wide field-of-view, low power display capable of operating untethered and manufacturable in quantity at a low unit cost.

3. A very difficult issue is position/orientation sensing. We need both high accuracy and long range, as well as the robustness necessitated by a fairly hostile factory environment. Ease of calibration and the ability to operate untethered are highly desirable.

4. Methods and systems for calibrating the display and sensor and registering the user are another critical issue. We require easy and quick ways to place the workpiece, the position/orientation sensor, the virtual screens of the display, and the user's eyes into the same coordinate system. Furthermore, because the user might jostle the display or it might slip out of position, we need a means to quickly detect that the user is out of registration, as well as ways to quickly re-register. For the sake of acceptance by factory workers, these registration procedures should be quick and convenient.

Our earliest efforts at calibration and registration techniques were rather clumsy and tedious, involving placing the user's eye in a known location, and then directly manipulating the calibration parameters until a virtual image and a physical image coincided. However, because of the large number of parameters, and the interaction between parameters, it was very difficult for an untrained user to determine correct values for all the parameters.

Over the past year, we have been prototyping and experimenting with some registration and calibration methods that are much more convenient, yet still retain good accuracy characteristics. They are the subject of this paper.

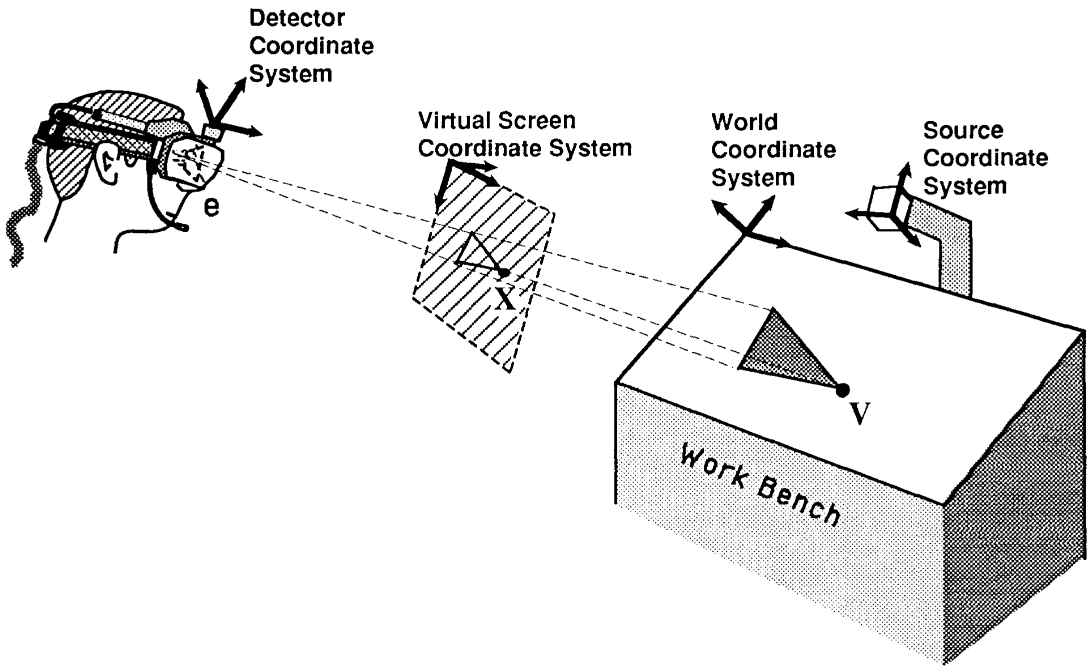

Figure 1

Definitions and Notation

Figure 1 illustrates the components of the system. A point **V**, fixed on the workpiece, is projected onto the virtual screen point **X**. Several coordinate systems are also shown. All coordinate systems are *a priori* right handed.

Objects, such as the triangle on the workpiece in Figure 1, are defined in *world* coordinates. The choice of origin and orientation of the world coordinate system is arbitrary, and may be chosen to match existing object definitions or to facilitate data entry.

Conceptually, the position sensor consists of a source that is rigidly attached to the world coordinates and a detector that is rigidly attached to the user's head. As the user moves his head, the position and orientation of the detector change with respect to the source. The position tracker therefore defines two additional coordinate systems: *source* coordinates and *detector* coordinates.

The virtual screen coordinate system is the frame defined by the location of the virtual screen and is fixed with respect to the optics of the physical display.

A column vector **V** in coordinate system C is expressed as $\mathbf{V}_c$. **V** in world coordinates is expressed as $\mathbf{V}_w$, in source coordinates as $\mathbf{V}_s$, in detector coordinates as $\mathbf{V}_d$, and in virtual screen coordinates as $\mathbf{V}_v$. The cartesian components of **V** are expressed as V_x, V_y, and V_z.

A transformation from one coordinate system to another consists of a pure rotation followed by a pure translation. This may be represented using a 4x4 homogeneous transform matrix with 6 degrees of freedom. A matrix $\mathbf{T}_{cd}$ transforms a vector expressed in coordinate system D to the same vector expressed in coordinate system C: $\mathbf{V}_c = \mathbf{T}_{cd} \cdot \mathbf{V}_d$

Mathematical Foundation

To correctly render a vector **V**, it is necessary to determine the line connecting **V** with the user's eye **e.** Illuminating the pixel that corresponds to the point at which this line intersects the virtual screen causes **V** to appear in the correct location in the user's field of view. To determine this point, the endpoints must be expressed in virtual screen coordinates. Let $\mathbf{X}_v$ be the point of intersection in virtual screen coordinates. One may compute $\mathbf{X}_v$ given $\mathbf{e}_v$ and $\mathbf{V}_v$,. Ignoring distortions introduced by the optics, one may compute the pixel on the physical screen to illuminate by scaling the x and y components of $\mathbf{X}_v$ by constants P_h and P_v, which represent the horizontal and vertical pixel resolution (in pixels per inch) of the virtual screen. Since $\mathbf{X}_v$ is located on the virtual screen, the z component is 0. Therefore, in order to compute the correct pixel to illuminate, we need to know the values of $\mathbf{e}_v$, $\mathbf{V}_v$, P_h and P_v.

Once the system has been fully calibrated, the following parameters are known or can be computed:

- $\mathbf{T}_{sw}$, the matrix that transforms a vector from world coordinates to source coordinates. This matrix is a function of the location and orientation of the source of the position sensor relative to the world coordinate system. If the source moves with respect to the world coordinate system, $\mathbf{T}_{sw}$ must be recomputed. This matrix is sometimes referred to in the literature as the alignment matrix or alignment correction.

- $\mathbf{T}_{vd}$, the matrix that transforms a vector from detector coordinates to virtual screen coordinates. This matrix is a function of the display optics and the position and orientation of the detector with respect to the head-mounted display. If the optics change or the detector mounting moves with respect to the optics, $\mathbf{T}_{vd}$ must be recomputed.

- $\mathbf{e}_v$, the location of the user's eye in virtual screen coordinates. This point will change if the user moves the head-mounted display with respect to his eye. It also changes from user to user. In the model used, we assume the eye is a pinhole imaging system. Since the field of view of the Augmented Reality systems is generally small, and the user will generally be looking forward at the workpiece, this is a reasonable assumption.

- P_h and P_v, the horizontal and vertical pixel resolution of the virtual screen. Note that these constants are not the same as the pixel resolution of the physical screen since the optics introduce additional scaling. These constants convert a point expressed in virtual screen coordinates on the virtual screen to pixel addresses on the physical screen, baring optical distortion.

At render time, the following additional parameters are known:

- $\mathbf{T}_{ds}$, the matrix that transforms a vector from source coordinates to detector coordinates. This matrix is a function of the position of the detector relative to the source, and can be computed based on the values returned by the position sensor (position and Euler angles of the detector with respect to the source, for example).

- $\mathbf{V}_w$, the location of the point to be rendered in world coordinates.

Given these values, it is possible to compute $\mathbf{V}_v$, the location of the point in virtual screen coordinates:

$$\mathbf{V}_v = (\mathbf{T}_{vd} \cdot \mathbf{T}_{ds} \cdot \mathbf{T}_{sw}) \cdot \mathbf{V}_w$$

Given $\mathbf{e}_v = (e_x, e_y, e_z)$ and $\mathbf{V}_v = (v_x, v_y, v_z)$ and the above conditions, it is possible to determine the point on the virtual screen to illuminate $\mathbf{X}_v = (x, y, z)$ by solving the following set of parametric equations:

$$x = (v_x - e_x)\, t + e_x$$
$$y = (v_y - e_y)\, t + e_y$$
$$z = (v_z - e_z)\, t + e_z$$

Since $\mathbf{V}_v$ and $\mathbf{e}_v$ are in virtual screen coordinates, we know the line connecting the two points must intersect the virtual screen at $z = 0$. Therefore:

$$t = \frac{e_z}{e_z - v_z}$$
$$x = (v_x - e_x) \cdot \frac{e_z}{e_z - v_z} + e_x$$
$$y = (v_y - e_y) \cdot \frac{e_z}{e_z - v_z} + e_y$$
$$z = 0$$

x, y, and z are now in virtual screen coordinates. The address of the pixel to illuminate is $(x \cdot P_h, y \cdot P_v)$.

Calibration

The calibration parameters are logically separable into three types: those associated with the position sensor, those associated with the virtual screen, and those associated with the user. The sensor parameters consist of the six degrees of freedom associated with the $\mathbf{T}_{sw}$ matrix (three translational, three rotational). Virtual screen parameters consist of the six parameters of $\mathbf{T}_{vd}$ and, in the simplest case, P_v and P_h. If the optics of the virtual display device introduce distortions in the virtual screen, they must be accounted for also. This discussion assumes minimal distortions. The only parameter which changes from user to user is $\mathbf{e}_v$, the location of the user's eye in virtual screen coordinates. Each time the user places the head-mounted display on his head, the location of the eye could potentially change with respect to the virtual screen. It is therefore only necessary to determine 3 parameters (e_x, e_y, and e_z) each time the user begins operation of the system. The other 14 parameters (6 for $\mathbf{T}_{sw}$, 6 for $\mathbf{T}_{vd}$, and P_h and P_v) only need to be recomputed when the physical system changes (e.g. the source moves, the optics change, etc.). It may also be necessary to adjust $\mathbf{e}_v$ during use to account for shifting of the head-mounted display on the user's head.

In our research, we have used two primary methods to determine the calibration parameters: direct measure and optimization methods. Direct measure involves using precision measurement equipment to directly compute the values of the calibration parameters. Optimization methods involve collecting statistical data on the location of the projection of calibration points from the physical world onto the virtual screen and minimizing an error measure that is based on the calibration parameters.

Direct Measure of $\mathbf{T}_{sw}$

If the manufacturer of the position sensor has reported the origin and orientation of the source coordinates with sufficient accuracy, it is possible to measure $\mathbf{T}_{sw}$ directly using a ruler and protractor. Simply compute the offset and orientation of the source coordinate system relative to the world coordinate system. However, we have found that the actual origin and orientation of the coordinate system for off-the-shelf position sensors are not accurately aligned with the stated

origin and orientation. Also, the origin is usually inside a physical object (the source), making it difficult to measure offsets to the origin accurately.

Another method of computing $\mathbf{T}_{sw}$ is to fix the source in the world coordinate system and record the position of the detector when it is at the origin, along the x axis, and along the y axis of the world coordinate system. Let **R** be the position of detector at the origin, **X** be the position along the x axis, and **Y** be the position along the y axis. Now compute the normalized vectors:

$$\mathbf{X}' = \frac{\mathbf{X} - \mathbf{R}}{\|\mathbf{X} - \mathbf{R}\|} \qquad \mathbf{Y}' = \frac{\mathbf{Y} - \mathbf{R}}{\|\mathbf{Y} - \mathbf{R}\|} \qquad \mathbf{Z}' = \frac{\mathbf{X}' \times \mathbf{Y}'}{\|\mathbf{X}' \times \mathbf{Y}'\|}$$

$\mathbf{T}_{sw}$ can now be computed. Note that **X'**, **Y'**, **Z'**, and **R** are column 3-vectors, so $\mathbf{T}_{sw}$ is a 4x4 matrix.

$$\mathbf{T}_{sw} = \begin{bmatrix} \mathbf{X}' & \mathbf{Y}' & \mathbf{Z}' & \mathbf{R} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

One can achieve better results by taking several measurements and averaging. Also, the above calculation does not guarantee that $\mathbf{T}_{sw}$ is a valid homogeneous transform matrix. One must make the rotational part (the upper left 9 elements) of $\mathbf{T}_{sw}$ orthogonal in a post-processing step.

A similar problem exists using this method as the method discussed above; namely, that the origin of the detector is generally at some unknown location within the solid detector. In order for the above calculations to work, one must measure **R**, **X**, and **Y** while the origin of the detector is directly on the origin, x axis and y axis of the world coordinate system. Since this is not generally possible, one must measure the values while a fixed point on the surface of the detector is on the origin and x and y axes and then adjust the results of **R**, **X**, **Y**, and **Z** by the fixed offset from the point used to the origin of the detector.

Direct Measure of $\mathbf{T}_{vd}$, P_h and P_v

Because the virtual screen is not directly accessible, direct measurement of $\mathbf{T}_{vd}$, P_h and P_v involves much more effort than of $\mathbf{T}_{sw}$. If one is intimately familiar with the optics of the head-mounted display, and one can compute the location of the detector attached to it, one can, in principle, compute $\mathbf{T}_{vd}$, P_h and P_v. However, such a procedure requires extreme care and precision, and it is unlikely to be feasible.

Another method we have investigated is the use of camera metrology to calibrate various parameters including $\mathbf{T}_{vd}$, P_v, P_h, and $\mathbf{e}_v$. Camera metrology involves first computing several parameters for one or more cameras using fixed calibration images [Sid-Ahmed & Boraie, Tsai]. Once this has been accomplished, one can then compute the location relative to the calibration image of any object in the field of view of the cameras. One could place a camera in the approximate location of the user's eye and take a picture of the virtual screen containing (for example) coordinate axes of known length in pixels. Using the known camera parameters, it is possible to compute the location of the virtual screen. If one records the location and orientation of the detector at the same time, then it is possible to compute $\mathbf{T}_{vd}$. Since the number of pixels in the image is known, and the actual size on the virtual screen can be computed from the captured images, one can also determine P_v and P_h by simply dividing the number of pixels by the actual size.

Care must also be taken when selecting camera calibration and image processing algorithms. Since the resolution of a standard video camera is at best about 500 x 500 pixels, the field of

view must be restricted and sub-pixel accuracy must be achieved to provide sufficient accuracy. Multiple cameras could also boost precision.

Direct Measure of $\mathbf{e}_v$

One method to determine $\mathbf{e}_v$ is to require that the user place his eye in a known location. Although this method is feasible, it is inconvenient for the user, and therefore unsuitable for use in a factory environment. We have instead elected to use a dual camera metrology method to locate the user's eye. The user simply looks towards a set of cameras such that his eyes are in the field of view of both cameras. The computer captures the image, performs some image processing to locate the center of the eye, and converts this point to virtual screen coordinates. Of course, this method will only work if the camera can discern the eye behind the head-mounted display.

Our current implementation images the surface of the pupil, which is not exactly the correct location of $\mathbf{e}_v$. Rather, $\mathbf{e}_v$ should be the nodal point of the eye. However, since the nodal point is almost directly behind the surface of the pupil, and we assume the user is looking generally forward, the error in projection should be small.

Optimization

Our success with direct measure has been limited. Also, direct measure tends to involve many different procedures and algorithms, all with their own problems and idiosyncrasies. A different approach is to attempt to collect data on the system and minimize some error measure that is a function of the parameters in question. We have tested one such method, and we are working to extend and simplify the procedure.

To calibrate the parameters of the HMD, the user looks through the display at a physical registration device with known geometry. The user then manipulates a cross-hair on the screen and moves his head until the cross-hair and an endpoint of the registration object are aligned. Note that in this phase of calibration, the cross-hair is not head-tracked. It stays in a fixed position on the virtual screen. Once the cross-hair is aligned with the registration point, the user pushes a button, and the location and orientation of his head at that time are recorded ($\mathbf{T}^i_{ds}$), along with the location of the cross-hair on the physical screen. The x and y positions of the cross-hair on the physical screen are divided by P_h and P_v to obtain the position of the cross-hair on the virtual screen. ($\mathbf{Y}^i_v$). Many such points are recorded from many viewing angles. Given current values (guesses) of the calibration parameters ($\mathbf{T}_{vd}$, $\mathbf{T}_{sw}$, $\mathbf{e}_v$, P_h, and P_v), the position and orientation of the user's head ($\mathbf{T}^i_{ds}$), and the position of the calibration point in world coordinates ($\mathbf{V}_w$), it is possible using the procedure outlined in the **Mathematical Foundations** section above to calculate where the computer predicts that the calibration point will appear on the virtual screen ($\mathbf{X}^i_v$). An error measure of the goodness of the current guess is the sum over all viewing positions of the difference between where the computer predicts the point should appear and where the user has indicated it appears:

$$\varepsilon = \sum_{i=0}^{N} | \mathbf{X}^i_v - \mathbf{Y}^i_v |$$

A gradient descent or other optimization method iterates the procedure, varying the parameters until ε is minimized. Since the calibration parameters need only be computed when the system changes, it is not necessary for the minimization routine to operate in real-time.

There are several pitfalls to this method. First, a representation of the transforms must be chosen. Since the rotational part must remain orthogonal, it is not possible to represent the transforms simply as matrices. One natural choice is Euler angles; however, Euler angles have degenerate cases, and a single orientation can be represented by a family of possible Euler angles. The algorithm we implemented uses quaternions to represent rotations. Quaternions are numerically more stable, although use of quaternions does introduce the additional necessity of renormalization of the values to keep them bounded. See [Funda, Taylor, and Paul].

Errors in the position/orientation detector will propagate to errors in $\mathbf{X}_v$, causing non-optimal values for the calibration parameters to be chosen. In order to simulate the effect of noise, we generate random head orientations and positions and compute the values of $\mathbf{X}_v$ and $\mathbf{Y}_v$ given a fixed set of calibration parameters. Noise is then be applied to the data, and the calibration optimization routines executed. The result is a different set of calibration parameters which are optimal over the noisy data. Using the new set of data, the projections are performed and compared with the noiseless case. Using this method, we have found that the errors in projection scale approximately linearly with errors in the position detector.

Another potential problem is that the formulation used has a degenerate solution. Let $\mathbf{t}$ be a column 3-vector formed from column 4 (the translational part) of $\mathbf{T}_{vd}$ and c be any constant. The following transformation is an identity, causing no change in the error measure ε:

$$
\begin{aligned}
P_h' &= c\, P_h \\
P_v' &= c\, P_v \\
\mathbf{e}_v' &= \mathbf{e}_v / c \\
\mathbf{t}' &= \mathbf{t} + \mathbf{e}_v\, (1-c) / c
\end{aligned}
$$

Geometrically, the degeneracy represents scaling the virtual screen while at the same time moving the eye towards the origin of the virtual screen and moving the virtual screen toward the detector.

A simple way to overcome this degeneracy is to assign an arbitrary value for P_v and solve only for the aspect ratio of the virtual screen, P_v / P_h. This reduces the optimization dimensionality by one and removes the degeneracy.

We have achieved good results with this optimization method, although it is quite dependent on good initial guesses and sufficient data to fully constrain the solution. Also, the task of collecting the data is somewhat tedious. Approximately 20 calibration points were viewed from several angles and distances. P_v was fixed at an arbitrary value (50 pixels/inch), and the program solved for the other calibration parameters.

Once we are able to use camera metrology to image the virtual screen (as described in the section Direct Measure of $\mathbf{T}_{vd}$, above), we will replace the human eye with a camera. Use of the camera will help automate the procedure.

Ultimately, we view the ideal solution to the calibration problem to be a combination of optimization methods and direct measure. Direct measure will be used to provide the optimization with a good initial guess. The optimization algorithm will then compute all calibration parameters (including $\mathbf{e}_v$, in this case the location of the camera). Since only $\mathbf{e}_v$ changes from user to

user, it is not necessary to repeat the calibration of the other parameters unless the HMD or the position sensor's source change. When a user begins to work in the morning, he would look into a pair of cameras, and the dual camera metrology algorithm would compute $\mathbf{e}_V$. If, during the course of his work, the HMD moves on his head or he notices that calibration is off, he again looks at the cameras to capture the new value of $\mathbf{e}_V$.

Results

The most useful accuracy measure of an Augmented Reality system is the error in the projected image relative the real image. For Boeing's touch labor applications, the projected image should appear at approximately arm's length. A simple calculation shows that, assuming the calibration parameters are exact, the projected error scales linearly with the positional error in the position/orientation sensor and as the sine of the error in the orientation times the distance to the object. We are currently using the Polhemus Isotrak unit, which uses AC magnetic fields to measure position and orientation. Any conductors or EM fields in the area affect the reported positions and orientations. At best, we have achieved about 0.2" and 2 degree accuracy, leading to a projection error of about 0.8" at 18". This represents a worst case displacement caused only by the position sensor. Newer position sensors have higher accuracy and improved robustness.

Direct measurement has achieved no more than about 2" accuracy in the projected image at arm's length, meaning that the virtual image may be displaced by as much as two inches from the physical image it represents. With the estimate that at least 0.8" of this error comes from inaccuracies in the position sensor, 1.5" of the error is caused by inaccuracies in the calibration parameters.

Direct measure of $\mathbf{e}_V$ using Dual Camera Metrology has achieved accuracy of better than 0.05" using two standard 640x480 video cameras whose field of view covers only the user's face. This level of accuracy was achieved using a linear algorithm without taking into account the non-linear aberrations of a lens system [Sid-Ahmed and Boraie]. Calibration images and the user's eye were digitized to sub-pixel accuracy.

Optimization achieves a much higher degree of precision. With a well calibrated system, we have achieved projection errors of order 0.5", the approximate resolution of the position detector. Because the error measure ε is a function of the errors in the position sensor, it is necessary to collect a large amount of data so that the error is averaged. With better position sensors, better results can be expected.

Conclusion

Augmented Reality demands much more accuracy than immersive VR, both from the position/orientation sensing system, and from the methods used to calibrate the sensor, display, and user. We have achieved the best results by separating calibration of the display and sensor, which will be performed relatively infrequently, from the registration of the user's eyes in virtual screen coordinates -- which would be done every time a user puts on the display. We have developed a set of direct measurement techniques and optimization algorithms for HMD, sensor, and workpiece calibration. While the optimization-based methods are tedious, they achieve high accuracy. For registration of the user's eyes, we have invented and experimented with a dual camera metrology/image processing method that achieves high accuracy and is much more convenient for the user. We expect this combination of calibration approaches to prove to be feasible for use in the factory environment.

Acknowledgments

The authors would like to thank the many people who have contributed to the AR project at Boeing, including (among others) Karl Embacher, Chris Esposito, Roman Fresnedo, Tom Grandine, Tom King, Vicki Lane, Henry Sowizral, Bruce Wilson, and Karel Zikan.

References

T. P. Caudell and D. W. Mizell, "Augmented Reality: an application of heads-up display technology to manual manufacturing processes," HICSS 1992

I. Sutherland, "A Head-Mounted Three Dimensional Display," *Proc. FJCC*, pp. 757-764, Thompson Books, Washington, D.C. 1968

M.A. Sid-Ahmed and Mohamed T. Boraie, "Dual Camera Calibration for 3-D Machine Vision Metrology", *IEEE Trans. Instr. and Meas.*, vol. 39, no. 2, pp 512-516, June 1990.

J. Funda, R. H. Taylor, and R. P. Paul, "On Homogeneous Transforms, Quaternions, and Computational Efficiency", *IEEE Trans. Robotics and Automation.*, vol. 6, no. 3, pp 382-388, June 1990.

H. Goldstein, *Classical Mechanics*, Addison-Wesley, Reading, Massachusetts, 1980, pp 158-165.

T. Furness, "Helmet-Mounted Displays and their Aerospace Applications", National Aerospace Electronics Conference, Dayton, OH, May 1969.

W. Robinett and J. Rolland, "A Computational Model for the Stereoscopic Optics of a Head-Mounted Display", *Presence*, vol. 1, no. 1, pp 45-62, Winter 1992.